

(19) World Intellectual Property Organization  
International Bureau(43) International Publication Date  
2 August 2001 (02.08.2001)

PCT

(10) International Publication Number  
WO 01/55889 A1(51) International Patent Classification: G06F 17/00,  
H04K 1/00

(21) International Application Number: PCT/US01/02609

(22) International Filing Date: 25 January 2001 (25.01.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
60/178,028 26 January 2000 (26.01.2000) US  
09/563,664 2 May 2000 (02.05.2000) US(71) Applicant (for all designated States except US): DIGI-  
MARC CORPORATION [US/US]; 19801 SW 72nd Av-  
enue, Suite 250, Tualatin, OR 97062 (US).

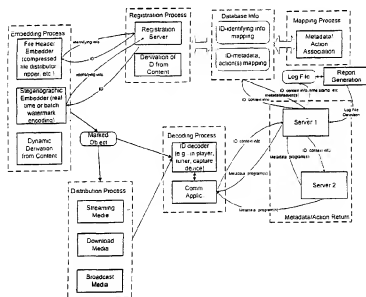
(72) Inventors; and

(75) Inventors/Applicants (for US only): MEYER, Joel, R.

[US/US]; 2742 SW Leah Court, Portland, OR 97219 (US).  
RHOADS, Geoffrey, B. [US/US]; 2961 SW Turner Road,  
West Linn, OR 97068 (US). LEVY, Kenneth, L. [US/US];  
110 N.E. Cedar Street, Stevenson, WA 98648 (US).(74) Agent: STEWART, Steven, W.; Digimarc Corporation,  
19801 SW 72nd Avenue, Suite 250, Tualatin, OR 97062  
(US).(81) Designated States (national): AE, AG, AL, AM, AT, AU,  
AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ,  
DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR,  
HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR,  
LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ,  
NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM,  
TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.(84) Designated States (regional): ARIPO patent (GH, GM,  
KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian  
patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European  
patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,  
IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF,  
CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: CONNECTED AUDIO AND OTHER MEDIA OBJECTS



(57) Abstract: Media objects are transformed into active, connected objects via identifiers embedded into them or their containers. In the context of a user's playback experience, a decoding process extracts the identifier from a media object and possibly additional context information and forwards it to a server. The server, in turn, maps the identifier to an action, such as returning metadata, re-directing the request to one or more other servers, requesting information from another server to identify the media object, etc. The linking process applies to broadcast objects as well as objects transmitted over networks (1251, 1252) in streaming and compressed formats.

**Published:**

- with international search report
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

- 1 -

## Connected Audio and Other Media Objects

### Technical Field

The invention relates to linking audio and other multimedia data objects with metadata and actions via a communication network, e.g., computer, broadcast, wireless, etc.

### Background and Summary

Developments in network technology and media content (e.g., images, video, and audio) storage, delivery, and playback are re-shaping the entertainment, information technology, and consumer electronics industries. With these developments, there are an increasing number of applications for associating media content with auxiliary data. The auxiliary data may provide information describing the content, copy control information or instructions, links to related content, machine instructions, etc. This auxiliary data is sometimes referred to as metadata. In many applications, metadata suffers from the drawback that it is vulnerable to becoming separated from an associated media signal.

Steganography provides a way to embed data in the media signal. As such, it offers an advantage over conventional ways to associate metadata with media signals. Examples of steganography include digital watermarking and data glyphs. Exemplary watermarking techniques suitable for still image and video content are shown in U.S. Patents 5,862,260 to Rhoads and 5,915,027 to Cox. Exemplary watermarking techniques suitable for use with audio content are shown in the just-cited Rhoads patent, as well as patents 5,945,932 to Smith and 5,940,135 to Petrovic

Advances in computer and wireless networking, multimedia coding, and higher bandwidth communication links are creating many new ways to distribute and enjoy multimedia content, such as music and movies. Coding formats for audio like MPEG Layer 3 (MP3) have already caused significant changes in music delivery to consumers. Despite the advances in technology, content distributors and broadcasters still need to address how to effectively promote and sell content.

This document describes systems and processes for linking audio and other multimedia data objects with metadata and actions via a communication network, e.g.,

- 2 -

computer, broadcast, wireless, etc. Media objects are transformed into active, connected objects via identifiers embedded into them or their containers. These identifiers can be embedded by the owner or distributor of the media object, or automatically created from the media object. In the context of a user's playback experience, a decoding process extracts the identifier from a media object and possibly additional context information and forwards it to a server. The server, in turn, maps the identifier to an action, such as returning metadata, re-directing the request to one or more other servers, requesting information from another server to identify the media object, etc. If the identifier has no defined action, the server can respond with an option for the user to buy the link and control the resulting action for all objects with the current identifier. The linking process applies to broadcast objects as well as objects transmitted over networks in streaming and compressed file formats.

Further features will become apparent with reference to the following detailed description and accompanying drawings.

## Brief Description of the Drawings

Fig. 1 is a diagram illustrating examples of media object linking processes and systems.

Fig. 2 is a diagram illustrating media object linking applications.

Fig. 3 is a diagram illustrating an operating environment for multimedia content management and delivery applications.

## Detailed Description

### ***Linking Audio and Other Media Objects via Identifiers***

The following sections describe systems and processes for linking audio and other media objects to metadata and actions via an identifier. For the sake of illustration, the disclosure focuses on a specific media type, namely audio signals (e.g., music, sound tracks of audio visual works, voice recordings, etc.). However, these systems, their components and processes apply to other types of media signals as well, including video, still images, graphical models, etc. As described further below, an identifier attached to an audio signal is used to connect that signal with metadata and/or

- 3 -

programmatic or device actions. In the context of this document, the terms "media object" and "audio object" refer to an electronic form of a media signal and audio signal, respectively. The linking of media signals applies to objects that are transmitted over wire networks (such as a computer network), wireless networks (such as a wireless telephone network), and broadcast (AM, FM, digital broadcast, etc.).

There are a number of ways to associate an identifier with an audio object. One way to associate the identifier is to insert it in the form of a numeric or alphanumeric code (e.g., binary or M-ary code) in the electronic file in which the audio is stored. Another way to associate the identifier is to embed it as auxiliary data in the audio signal using steganographic methods, such as digital watermarking or other data hiding techniques. Yet another way is to derive the identifier from the audio signal, the table of contents, the file system structure, or its container (e.g., an electronic file or physical package for data like flash memory, Digital Versatile Disk (DVD), minidisk, or compact disk (CD)). The physical media may have identifying characteristics, such as a unique identifier or encoded metadata, or other attributes from which an identifier can be derived (e.g., CD disk wobble).

When the identifier is associated with metadata or actions, it transforms the media object into a "linked" object. The identifier travels with the object through distribution, including in some cases, through physical distribution in packaged media and through electronic distribution (broadcast or network communication). The identifier may travel within the same band as the audio object, such as a watermark, or via a separate band, such as a file header or footer or separate broadcast band. A decoding device or programmatic process extracts the identifier from the object and uses it to retrieve related data or actions ("metadata"). In the case of an audio object, like a song, the metadata typically includes the title, artist, lyrics, copyright owner, sound recording owner, information about buying or sampling opportunities and URLs to this type of data as well as web sites and other programs and devices. Linked actions include device or programmatic processes for electronically establishing a license, transferring content (either streaming or download), sending an email, recording marketing data about a transaction, etc. The identifier allows a fan of a particular type of music or artist to get more information about the music and to buy more music.

- 4 -

From the perspective of the artists and record labels, the identifier provides an additional opportunity to promote their music and sell content, concert tickets, etc.

In addition, in some implementations where identifier-linking transactions are monitored, it enables the vendors of music to gather data about electronic transactions triggered by the link. For example, users of information may choose to provide information about themselves when they register their decoding device or software with the system. A user ID or other context information may then be recorded when the identifier is extracted and used to trigger a transaction. Many entities involved in the distribution of media signals can benefit from the linking capability. Artists can link their music to information about themselves and provide electronic buying opportunities for music, concert tickets, clothing, etc. Rights holding organizations can use the link to inform users about itself and licensing opportunities. In some cases, the link may also be used to monitor playing and distribution of copies of the music. Record labels can link their music to information about the artist, the label, electronic buying opportunities, etc. Electronic retailers can increase sales by linking users to opportunities to sample and buy additional music (via download or streaming delivery over a wire or wireless network). Conventional brick and mortar retailers can use linking to provide information about the music and to provide buying opportunities. Radio stations and other broadcasters can use the linking capability to bring users to their web sites, creating advertising revenue, to provide electronic buying opportunities for music, concert tickets, clothing items, etc. These and other forms of linked metadata and actions may be implemented in various combinations in different application scenarios.

Depending on the application, the identifier may identify the media object, in which it is embedded, or entities, things or actions other than that particular media object. One type of identifier is an object ID that identifies an audio object. This identifier may be a number associated with the object, such as its International Standard Recording Code (ISRC). Another type of identifier is distributor ID that identifies the distributor of the audio object. Another type of identifier is a broadcaster ID that identifies the broadcaster of the audio object. Of course, more than one identifier may be encoded into an audio object or its container. In the event that an object ID is not encoded with an audio object, but instead, a distributor or broadcaster identifier is

- 5 -

encoded with the object, other context information, such as the time of play back or distribution, location of distribution, etc. may be used to identify the audio object as part of the linking process. An example is a radio station that marks its broadcasts with a station ID and maintains a playlist database with the air times of each audio object.

- 5 At decoding time, the station ID is extracted and used along with context information such as the air time of the audio object to look up the audio object or its corresponding metadata and actions. This approach enables the linking system to provide audio object specific metadata or actions even without requiring a unique object identifier in every audio object.

## 10 **System Implementation**

- Fig. 1 is a diagram of a system configuration of linked media objects. In this configuration, an identifier links audio objects to metadata via an electronic network, such as the Internet, a wireless network, or a broadcast network. As depicted in Fig. 1, an embedding process may be used to encode an identifier in an audio object or its
- 15 container. In some cases, an embedding process encodes the identifier in the audio file (e.g., a tag in a file header or footer), in the audio signal (a digital watermark), or in the physical packaging. The identifier may also be derived as a function of the audio signal or other information in the file or physical packaging (e.g., track information on a CD). In the case of dynamically derived identifiers, an embedding process is not necessary
- 20 because the identifier can be derived from the content at decoding time.

- In some application scenarios, the embedding process interacts with a registration process to get an identifier. The embedding process provides information about the object (e.g., a title and artist name, an ISRC, name of distributor, etc.). In response, the registration process provides an identifier and stores a database record of
- 25 the association between identifier and the object or other information used in decoding to identify the object, such as its distributor or broadcaster. The registration process may be used to assign an identifier to an audio object and to distributors or broadcasters of audio objects. The embedding and registration processes may occur before the audio object is distributed to consumers, or sometime thereafter, such as when a user transfers
- 30 (e.g., "rips") a media object from one format to another (e.g., a packaged format to an electronic file format such as a compressed file format).

- 6 -

Once registered, an interactive or automated mapping process associates the identifier with data or actions. The registration process creates a database of identifiers and associates the identifiers with corresponding media objects, distributors, broadcasters, etc. The mapping process associates the identifiers with corresponding metadata or actions.

Once associated with an audio object and metadata, the identifier transforms the audio object into a linked object. The identifier remains with the object through distribution, although some embedding processes are more robust than others to intentional or unintentional distortion/removal of the identifier. There are a variety of different distribution scenarios. Some examples depicted in Fig. 1 include transferring an audio object over a computer network, streaming the object over a computer network, or broadcasting it (e.g., AM/FM broadcasting, digital broadcasting, broadcasting over wireless carriers, etc.). Whatever the distribution process, a user ultimately receives the linked object in a player, tuner, or capture device.

To activate the linked object, a decoding process extracts the identifier and uses it to access associated data or actions. The decoding process may be implemented as a separate program or device, or integrated into a player, tuner, or some other capture device, such as a listening devices that converts ambient audio waves to an electronic signal and then extracts the identifier from the signal.

In the configuration shown in Fig. 1, the decoding process forwards the extracted identifier to a communication application, which in turn, forwards it in a message to a server. The decoding process or the communication application may add additional context information to the message sent to the server. The context information may relate to the user, the user's device, the attributes of the session (time of playback, format of playback, type of distribution (e.g., broadcast or transmitted audio file), etc.) Based on identifier and optional context information, the server determines an associated action to perform, such as re-directing an identifier or context data to another server, returning metadata (including programs, content, etc.), downloading content, logging a transaction record. To find the associated action or actions, the server maps the identifier to actions based on the information established in the mapping process. The server may: 1) look up the data and actions in a local database stored in its memory subsystem; 2) route the identifier to one or more other



- 7 -

servers via the network, which in turn look up related actions and data associated with the identifier; or 3) perform some combination of actions 1 and 2.

In the first case, server 1 returns data or actions associated with the identifier. The server may look up related data based on the identifier alone, or based on the  
5 identifier and other context information. Context information may be information provided by the user, by the user's computer or device, or by some other process or device. In the second case, the server looks up one or more addresses associated with the identifier and forwards the identifier and/or possibly other context data to secondary  
10 servers at these addresses via conventional networking protocols. Again, this context data may include data from the user, the user's computer, some other device or database. For example, server 1 might query a remote database for instructions about how to process an identifier. These instructions may specify data to return to the communication application or to forward to another server, which in turn, looks up  
15 associated data and returns it to the communication application. A server may return data that an audio player displays to the user or uses to control rendering of the content. For example, the server can tell the player that the object contains inappropriate content for children. The player or user can make decisions about whether or how to play the material based on this information.

Both the server and the player can adopt a set of rules. The server rules may be  
20 used to control what the server returns in response to an identifier and context data. The player rules may be used to control what the player displays to the user or how it renders the content based on data returned from a server.

Either the first server, or a server one or more levels of indirection from the identifier may return data and programmatic actions to a player via the communication  
25 application. Each server in these levels of indirection receives a database key, such as an identifier or context information, from the previous server, and uses it to look up corresponding actions. These actions may include returning data or programs to the communication application or to previous servers in the routing path of the message from the communication application. Also, the servers may route requests for  
30 information or actions to other servers. The server or servers may return data or perform actions in response to the identifier (or other context data) that do not directly impact the decoding process, or the device in which it operates.

- 8 -

The system depicted in Fig. 1 allows several different interested parties to establish services linked via the identifier. For example, server 1 can be configured to provide generic promotional and/or licensing information associated with an identifier. If the content owner, distributor, retailer, artist or other related party wishes to provide information or services for a connected object, then server 1 may also route the identifier for that object, and possibly context information, the address of the communication application, and instructions, to servers maintained by these entities. These servers, in turn, provide promotional, sales, or licensing information, and electronic buying or licensing opportunities specific to that entity back to the consumer over the network via the communication application.

In the context of a network configuration, Internet protocols may be used to return data to the communication application or to the device or system in which it operates. The communication application may be implemented in a web browser, such as Internet Explorer or Netscape Navigator. Examples of ways of exchanging information between a client player and a server include returning a web page with metadata and program scripts designed to run on the end user's system. The metadata itself may include active links, such as URLs to other network resources, such as a web site or some other network service. The path of the identifier from the decoding process, and the return path from a server to the communication application may include one or more hops through a wire or wireless connection using standard wire and wireless communication protocols like TCP/IP, HTTP, XML, WAP, Bluetooth, etc. In addition, data returned to the user may be routed through one or more servers that may forward the data, and in some cases, augment the data or modify it in some fashion.

Fig. 2 is a diagram illustrating applications of the system depicted in Fig. 1. In the application scenarios depicted in Fig. 2, an embedding process encodes an object identifier (OID) into an audio file, such as an ID3 tag in the header of an MP3 file or audio frame headers in the MP3 file. Fig. 2 shows two embedding scenarios. The first is an MP3 distributor that embeds OIDs in MP3 files before transmitting them over a network, such as the Internet, typically via a web site interface. The second is a file ripping process where a programmed computer or other device extracts an audio object from packaged media such as a CD and converts it into a coded file format like MP3.

- 9 -

In the latter case, the ripping process may extract metadata from the CD, such as the table of contents, and use this metadata as a key to a database (CDDb) to get information about the songs on the CD, such as title, artists, etc. The table of contents or other metadata from a package medium, such as optical or magnetic storage or flash memory, may be hashed into an index to a database entry that stores information about the media signal stored on the medium. The ripping process uses the information returned from the database to identify the audio objects on the packaged media so that they can be associated with an OID. This is an example of identifying information used to associate an OID with an audio object. As part of the coding process, the ripping process inserts the OID in the file header of the MP3 file.

Later, when a user opens or plays the marked MP3 in a player, such as a software player like the real player, Liquid Audio player, Windows Media Player (WMP), WinAmp, MusicMatch, etc., a plug-in software module in the player extracts the OID and forwards it to a server via an Internet connection. The plug-in may establish its own Internet connection, or pass the OID to an Internet Browser, which in turn, establishes a connection (if one is not already present) with the server. As an intermediate step, the plug-in may display a window with user options, such as "learn more about the song", "play the song", or both. The user can then choose to get more information by actuating the first or third options in the user interface window, which cause the plug-in to forward the OID to the server.

The server then returns a web page associated with the OID, or re-directs the OID to another server (e.g., one maintained by the content distributor or owner), which in turn, returns a web page of information about the object and links to related actions (e.g., a link to a licensing server, a link to a server for buying and downloading related music etc.). The licensing server may be programmed to download software players and new music offerings compatible with those players. For instance, the licensing server may provide software for decrypting, decoding, and playing electronically distributed music according to usage rules packaged with the electronically distributed music. In this application scenario, the linking of the MP3 file enables the content owner to market music and products that promote the sale of audio objects in other formats, included formats protected with encryption, watermark copy managements schemes, etc.

- 10 -

In the event that a media object is not linked, the decoding and server processes can be programmed to enable the user to purchase a link for the object. For example in one scenario, the player plug-in displays a graphic for a link information indicating that the link is available after determining that an OID is not in the file. If the user clicks on the graphic, the plug-in displays more information about the procedure for purchasing or renting a link. This information may be provided in conjunction with querying the server and displaying information returned from the server, or alternatively, providing pre-programmed information incorporated into the plug-in. If the user is interested in purchasing the link, he or she can then enter input (e.g., click on a button such as "Get Link") that initiates the process of registering an OID with the object and associating metadata or actions with the OID. The process of registering the OID and associating the OID with metadata or actions may be performed as described in this document. This scenario provides yet another mechanism for transforming content into connected content.

There are many possible variations to the applications scenarios illustrated in Fig. 2. During the file ripping process (or some other embedding process), the embedder may generate a unique ID from the metadata read from the packaged media on which the media object resides. One example of such an ID is the number derived from CD metadata currently used to index information in the CDDB database. This ID may then be embedded in the audio object or its file header/footer. During OID registration, the registration process may inform the embedding process that the OID (and thus, the object for which it was derived) has not been associated with metadata or actions. In this case, the user may be given an opportunity to purchase the link, either at the time of ripping, or in the future, wherever the object travels. In the latter case, the OID in the object is associated with an option to buy the link and customize the data and/or actions associated with that link. Rather than link to promotional information, the OID gives users an option to buy or rent the link and provides them with an opportunity to customize it (e.g., linking it to a custom web site). Once customized, other users that open or play the file will then be able to link to the customized information or actions.

- 11 -

To assert control over the type of customization that users may perform, the registration and mapping processes can place constraints on the types of metadata and actions that users can link to a media object.

In the multimedia content industry, there are typically many rights holders and entities involved in the distribution process. This may present a conflict when linking a media object to one entity. One way to address this problem is have an object link to many different entities. For example, the server could map an OID to many entities and return links to retailers, distributors, record labels and artists. Another way to address it is to encode additional information about the distributor in the OID. For example, the OID includes fields that identify the object and its distributor. If a user activates the link to purchase products, including media objects, then the distributor name is logged with the purchase and that distributor is credited with royalties associated with the transaction. The distributor field may also be used as a key to look up the appropriate action for the OID, such as re-directing the OID to the web server of the entity associated with that OID. In this approach, even if the OID directs a user to a record label's website, the distributor field can be used to credit the distributor with a royalty for the linking transaction.

The entity responsible for maintaining a web site linked via an identifier can make deals with online resources for providing data about a media object such as lyrics, song titles, radio station play lists. The website may link to this information, access it via a database manager, etc.

### ***File identifiers***

One form of identifier is an identifier that is inserted in an audio object file, but in a distinct field from the audio signal itself. Some examples are file headers and footers.

This file identifier may be assigned before or after distribution of the audio object to consumers. In addition, it may be derived from the audio signal or other information in the file. For example, an identifier generator may derive a unique or sufficiently unique identifier from a portion of a music signal. A variety of methods for generating a unique numbers based on a unique collection of numbers may be used.

- 12 -

The process of embedding a file identifier may be done at the time of encoding or transcoding a file. For example, the file identifier may be inserted during a ripping process, such as when a device or programmatic process converts a song from a format stored on packaged media, like a CD or DVD, to an electronic, and compressed form, such as MP3 or some other audio codec. As another example, the file identifier may be inserted when a device or programmatic process transcodes an electronic music file from one codec format to another. Yet another example is where a file is taken from a digital or analog uncompressed format, and placed in another format for distribution.

### ***Identifiers Embedded in Audio Signal***

Another way to associate an identifier with an audio signal is to embed the identifier in the audio signal using steganographic methods, such as digital watermarking or other data hiding techniques. Many of such techniques have been developed and are described in published articles and patents. Watermarking methods are described in US Patent application 09/503,881. Other examples of methods for encoding and decoding auxiliary signals into audio signals include U.S. Patent Nos. 5,862,260, 5,940,135 and 5,945,932.

The steganographic embedding method may be performed in a batch process. Consider a distributor of electronic music via the Internet or some other network, or a broadcaster of music such as a radio station. In each case, the distributor and broadcaster have a collection of audio objects. The embedding process may operate on this collection of objects in a batch process by retrieving an electronic version, encoding an identifier obtained from the registration process, and returning the marked version for later distribution or broadcasting. In some cases, it is desirable to do watermark embedding in an iterative process in a studio environment to encode the watermark with an intensity that achieves desired perceptibility and robustness requirements.

The steganographic embedding method may also be performed at the time of transmission of an electronic file or broadcast of the audio object. In the case of distribution via a network such as the Internet (e.g., streaming or file download), real time embedding enables the embedding process to also embed context information that is specific to the consumer (or the consumer's computer) that has electronically ordered

- 13 -

the object. For example, when the user requests a file in a streaming or a compressed file format via the Internet using her browser, the distributor's server can request information (perhaps voluntary) about the user to be associated with the transmitted object. Later, the decoding process or the servers that map the identifier to actions or metadata can use this information to determine the types of information to provide or responsive action to perform.

In the case of broadcasting, real time embedding enables the identifier to be steganographically embedded throughout an electronic version of the audio signal just before, or as part of the broadcasting process.

An object or distributor ID (as well as other identifiers or context information) can be embedded in the payload of a watermark that is also used for copy control. Portion of the watermark can be used to control whether the object can be played, transferred, recorded, etc., while another part can be used to carry identifiers and other metadata for linking functions described in this document. Alternatively, entirely separate watermark encoding and decoding methods may be used for copy control and linking functions.

A watermarking process may be used to encode different watermarks in the various channels of an audio signal. Message information may be embedded in one or more channels, while synchronization or orientation signals used to detect and decode the message information may be encoded in other channels. Also, different messages (e.g., different identifiers) may be encoded in different channels. At decoding time, the different identifiers can trigger different actions or link to different data.

In broadcasting applications, an identifier may be encoded along with the broadcast of the associated media signal by modulating a subcarrier of the main carrier frequency used to transmit the media signal. The subcarrier conveys auxiliary data such as the identifier, while the main carrier conveys the associated media signal. To reduce audibility of the auxiliary data (e.g., the identifier(s)) encoded in the sub-carrier, the data can be randomized by applying it to a pseudorandom or random number by some function that may be inverted in the decoding process, e.g., multiplication or exclusive OR functions. One example of sub-carrier encoding and decoding is Active HSDS 97 developed by Seiko Corporation.

**Identifiers in Digital Radio Broadcasts**

Some forms of digital radio broadcasts support transmission of metadata along with media signals. This metadata can also be used to carry one or more identifiers that are mapped to metadata or actions. The metadata can be encoded at the time of broadcast or prior to broadcasting. Decoding of the identifier may be performed at the digital receiver. In particular, the digital receiver receives the broadcast data, extracts the identifier, and either automatically, or at the user's direction, forwards the identifier to a server to look up the associated metadata or action.

**Dynamic Identifier Extraction from Audio Content or Related Data**

As noted above, another way to associate an identifier with a corresponding audio signal is to derive the identifier from the signal. This approach has the advantage that the embedding process is unnecessary. Instead, the decoding process can generate the identifier from the audio object. In this case, the decoder computes a fingerprint of the audio signal based on a specified fingerprinting algorithm. The fingerprint is a number derived from a digital audio signal that serves as a statistically unique identifier of that signal, meaning that there is a high probability that the fingerprint was derived from the audio signal in question. One component of fingerprint algorithm is a hash algorithm. The hash algorithm may be applied to a selected portion of a music file (e.g., the first 10 seconds) to create a fingerprint. It may be applied to discrete samples in this portion, or to attributes that are less sensitive to typical audio processing. Examples of less sensitive attributes include most significant bits of audio samples or a low pass filtered version of the portion. Examples of hashing algorithms include MD5, MD2, SHA, and SHA1.

As an aside, fingerprinting may also be used to determine whether an audio signal has been watermarked. The fingerprinting application can evaluate a fingerprint for a received object and compare it with one for a watermarked object (or unmarked object) to determine whether the object is likely to be watermarked. Certain fingerprints can be associated with certain types of watermark methods. Using the fingerprint, a decoding device can select an appropriate watermark decoding system for the object.



While specifically discussed in the context of audio objects, the fingerprinting process applies to other types of multimedia content as well, including still images, video, graphics models, etc. For still images and video, the identifier can be derived dynamically from a compressed or uncompressed version of the image or video signal.

- 5 The fingerprinting process may be tuned to generate a specific identifier based on the type of file format. For example, the process extracts the file format from the file (e.g., from a header or footer), and then uses a fingerprinting process tailored for that type of file (e.g., a hash of a compressed image or video frame). The dynamic identifier computed by this process may be associated with metadata and/or actions using the processes and systems described in this document.
- 10

### ***Registration Process***

- One way to implement the registration process is to build client and server application programs that communicate over a computer network using standard network communication protocols. The client may be implemented as a software program that provides identifying information about an audio object. It can obtain the information by prompting the user for the identifying information, or from extracting it from the audio object or its container. The server may be implemented as a database management program that manages identifiers and corresponding audio objects. When queried to provide an identifier for particular identifying information, the program checks whether it has already assigned an identifier to an object based on the identifying information. If so, it returns that identifier that has already been assigned. If not, it assigns a new identifier number, creates a new entry in the database for that number and its associated identifying information.
- 15
- 20

- 25 The type of identifier used to link audio objects varies with the application. As such, the registration process may vary as well. One type of identifier is a unique identifier for an audio object. Another type of identifier is one that identifies some attribute of the audio object, but does not uniquely identify it, such as a distributor or broadcaster identifier. This type of identifier requires additional context information to uniquely identify the audio object at the time of linking it to actions or metadata. For these types of identifiers, the registration process provides information identifying the
- 30

- 16 -

attribute of the audio object, such as its distributor or broadcaster. In response, the server provides an identifier that may be embedded in several audio objects that share that attribute.

One example is a broadcaster ID, such as a radio station ID. Audio broadcast  
5 by the radio station is embedded with this radio station ID. To identify the object, context information such as the play time captured at the tuner is used along with the radio station ID extracted from the received audio signal to identify the audio object. The decoding process forwards this information to a server. Using the radio station ID and context information, the server maps the ID to an appropriate action. This may  
10 include querying a radio station's playlist database for an object identifier based on the station ID and context information. The server can then map the object identifier to an action or metadata based on the object ID returned from the playlist database. Other scenarios are possible. For example, the server could forward the station ID, context data and decoder address to a radio station server, which in turn, looks up the  
15 appropriate action or metadata (e.g., web page) and sends it to the device that decoded the station ID.

Broadcast content can also be associated with object identifiers. One way to implement the identifier assignment process is to allocate a unique set of identifiers with each broadcaster/distributor. Those broadcasters or distributors are then free to  
20 assign the identifiers to media objects as they wish. Once they complete the identifier assignment process, they may then associate the identifiers with the metadata or actions in a mapping process.

### ***Embedding Process***

The embedding process may be integrated into a software program along with  
25 the client of the registration process described in the previous section. This integration of registration and embedding functions is particularly suited to a batch embedder, where processing time required to request an identifier is less of a concern.

In real time embedding, the identifier or identifiers are preferably available for associated audio objects before embedding begins. For example, the identifiers can be  
30 maintained in a local database on the embedding computer or device and indexed by

- 17 -

object title. Distributor and broadcast identifiers are more straightforward because they may be applied to several different audio objects.

The embedding process may also be implemented in an embedding clearinghouse system. The embedding clearinghouse is a computer or other electronic system that analyzes media objects and embeds one or more links in the media objects. The clearinghouse may be implemented in a server on a network, such as the Internet and operate on content in a "push," "pull," or some combination of push and pull models. In the push model, users and other systems send media objects to the embedding clearinghouse for analysis and embedding. The pull model, the clearinghouse has the capability to search for and gather media objects for embedding and analysis. One example of this pull model is an Internet search process called a spider that crawls the Internet, searching for media objects to analyze and embed with one or more identifying links.

The embedding clearinghouse analyzes a media object (perhaps based on out of band data like a file header or footer) and inserts an identifier. This identifier may link to a metadata and actions, such as re-direction to a web site offering products, services, and information related to the content. The embedding clearinghouse may incorporate search engine technology to execute a key word search based on information from the media object and then associate the media object with a series of related URLs returned from the Internet search. The process may be automatic, or with some user input to select which sub-set of links should be inserted.

The embedding clearinghouse may also offer an identifier embedding services for those wanting to link their media objects with metadata, actions, etc. In this application scenario, the embedding clearinghouse may be implemented as an Internet server that is accessible via a web page using conventional network communication and web protocols. To access the server, users visit a web page using an Internet browser. In exchange for a fee, which may be tendered electronically over the Internet from the user's computer to the server, the server provides an embedding service to embed an identifier into a media object uploaded from the user via the user's computer and Internet connection. The user can select the information to associate with a media object, such as generic identifying information (e.g., title, author, owner), generic licensing information, or special information or actions. The provider of the

- 18 -

embedding clearinghouse server hosts the generic information, while the special purpose information and actions are accessed through re-direction. In particular, the provider of the clearinghouse server links the embedded identifier to an address or set of addresses of servers that provide the special information or actions. Then at decoding time, the decoding process sends the identifier to the provider's server, which in turn, redirects the identifier to a secondary server or servers that provide special purpose information or actions (e.g., redirect to a web page of the content owner, download related content, provide electronic licensing services, etc.).

### ***Decoding the ID and Embedded Context Data***

The implementation details of the decoding process depend on how the identifier is encoded into an audio object or its container. In the case where the identifier is encoded in a file header or footer, the decoder may be a software program or digital hardware that parses the header/footer and forwards it to the communication application. One way to implement this type of decoder is to integrate it into a media player as a plug in program. Examples of media players include Windows Media Player from Microsoft, Liquid Audio player from Liquid Audio, Winamp, Real Player from Real Networks. Preferably, the plug-in gives the user visual feedback that the identifier has been detected and displays a window with options to access more information or actions available via the link. For example, the user can be presented with a user interfaces prompting the user to click for more information or buying opportunities. If the user selects these options, the plug-in forwards the user selections and identifier to the communication application, which forwards them to the server (e.g., server 1, Fig. 1).

In the case where the identifier is steganographically encoded in the audio object, a corresponding decoder extracts the identifier. This type of decoder may be implemented as a plug in to a software player as described in the previous paragraph. It may also be implemented in a tuner for broadcast content, or in a listening device that captures audio from the ambient environment.

In the case where the identifier is derived from the content or container metadata, the decoder captures the pertinent portion of the audio object, and generates

- 19 -

the identifier as described above. This type of decoder can be implemented in a software or hardware player, a tuner, etc.

The decoder may collect identifiers in response to a user request while objects containing these identifiers are being played. For example, when the user is playing music, he may like a song and want to buy it or get more information. This feature may be implemented by building an interface that has a button or voice recognition that enables the user to request information or a buy/license opportunity. Once captured, identifiers can be forwarded along with user instructions to the appropriate server.

However, one particularly useful feature is to enable the user to fetch information and make orders from music as the music is playing. The system described previously supports this feature because the decoding process can forward the identifier or identifiers, embedded context information, or additional context information (user information, play time, broadcast type, file type, player type, operating system type) to the communication application as the music is playing. The user can trigger the linking action by pressing a "fetch" button, or saying fetch to a voice activated input device that causes the decoding device to package a message and invoke the communication application (e.g., Internet browser). In turn, the communication application forwards the message to a server that parses the message and determines the associated action.

The activation of the "fetch it" feature may be made on a handheld device that communicates with a decoding device in a tuner via a wireless connection. For example, a user may press a button on a remote control device, like a key chain, which sends a wireless signal to a receiver in the tuner. The receiver invokes the decoding process. The tuner may also send metadata from the server to the remote control device for display using a similar wireless connection. Infrared or RF transceivers, for example, may be used to communicate the data back and forth.

The decoding device may also provide continuous decoding of identifiers. When the user requests a "fetch," the identifier and context information for the current song may be forwarded to the server. Also, the decoding device may automatically fetch generic information such as song title and artist so that this information is immediately available to the user.

Another possible implementation is to temporarily buffer identifiers extracted from some predetermined number of the most recent songs, titles, etc. These identifiers

- 20 -

can be stored along with other metadata, such as a time stamp, to inform the user when they were captured. The user can then select one or more of the items to send to the server for more information or related actions.

These features may be implemented in one or more devices. While the example above discusses a remote control device and a separate tuner with a decoder, these functions may be integrated into a single device, such as a car stereo, phone handset, personal digital assistant, and a variety of other types of players or tuners.

The identifier enables dynamic linking. Dynamic linking enables the identifier encoded with a media object to remain fixed, while the metadata or actions associated with that identifier can be changed. To change the associated metadata, the mapping process edits the identifier database to associate new metadata or actions with an identifier. The mapping process can be automated to change metadata or actions associated with an identifier at periodic intervals or in response to system events. In addition, a user may change the associated metadata or actions interactively at any time. To facilitate access to the database, a web-based interface can be added to the database.

Dynamically linked data returned from a server to a player environment can be displayed to the user in a variety of ways. One way is to display it in a web page or user interface window of a player. The data can be animated by scrolling it across the visual display. The data can also be displayed in the form of HTML links, which, when activated, cause the download of other data or initiate actions, such as playing streaming content from a server.

### **Server Types**

As discussed elsewhere, the servers used to link identifiers to actions may be programmed to provide a variety of actions including:

- returning data and HTML links (e.g., in the form of an HTML document, scripts, etc.)
- downloading media signals in streaming or file format
- performing an electronic transaction (selling products like CDs, DVDs, concert tickets, etc. via computer transaction using credit cards, digital money, etc.)
- establishing a license to use a linked media object

- 21 -

- re-directing to another server
  - performing database look up operations for related information, links, actions
  - performing database look up to uniquely identify a media object based on distributor/broadcaster ID and other context information
- 5      • creating a transaction log

This is by no means an exhaustive list. Another type of server action is to initiate a process of searching a database, a collection of databases or the Internet for additional information related to a linked media object. This type of search service  
10      may be performed continuously and the results associated with the identifier. Then, in response to a request from a decoding process, the server can return a digest of the results with links to web pages for additional information.

### ***Communication Application***

The implementation details of the communication application are highly  
15      dependent on the type of communication link and protocols used to connect the decoding process to a server. Above, an Internet browser is provided as an example. A browser may be implemented in conventional PCs, handheld devices, wireless phones, stereo systems, set top boxes, etc. However, the communication application need not be based on computer network protocols. For wireless devices, where the marked  
20      content is played on wireless carrier frequencies, the communication application can employ wireless communication technology to forward identifiers and context information to servers that map this information to actions or metadata and return it via a wireless carrier frequency to user's handset.

### ***Tracking Transactions and Report Generation***

As depicted in Fig. 1 and described above, the servers for mapping identifiers to  
25      actions may be programmed to dispense a transaction log into a log file. A report generation process can then enable users to define and request queries of data from the log file based on a particular identifier, a particular type of context information (time frame, geographic location, user demographics, etc.), a particular action, etc.

**Capture Devices**

As noted above, the decoding process may be implemented in a variety of devices or software that process media objects. These devices and software include programmable devices such as personal computers, personal digital assistants, telephone handsets, set-top boxes, personal stereos, hi-fi components, tuners, receivers, televisions, etc. as well as hardwired devices that may be incorporated into these systems and devices.

In some contexts, it is useful to implement a recording function. This is particularly true in devices that receive a broadcast or stream of media content and need to capture at least a portion of it to decode an identifier. Examples of these devices are radio receivers, and wireless telephone handsets. The record function may be automatic or user activated. In the latter case, the user actuates an input device to control the record process and optionally the record duration. For example, the user may hear a song that she likes and press record. The device, in turn, records at least a part of the object that is currently being received (an audio, visual or audio visual signal). The user can then decide contemporaneously or at a later time to execute the identifier decoding process on the recorded signal. The recording function can be designed to execute for a pre-determined or user specified duration.

In the case of radio and television tuners/receivers, the record function can be used to capture a media signal as it is received. In the case of a telephone handset, the record function can be used for a variety of functions, such as recording part of a telephone conversation, recording speech or other ambient audio through a microphone, or recording a media signal received by the handset via a wireless communication channel. The recordings can be compressed and stored in local memory on the device. In addition, they may be annotated with metadata about the media signal, such as a time stamp to show time of capture, a location stamp to show location of capture, metadata extracted from the object (in band or out of band data), etc. A global positioning device may provide the location stamp. Some wireless phone systems are capable of computing location of a telephone handset via triangulation. This location data may be used to provide geographic location coordinates or the name of nearby landmark, city name, etc.



The metadata may be displayed on a display device to help the user remember the context of a particular recording. In addition, it may be provided as context information along with an identifier to a server that links the identifier and context information to metadata or actions.

5

### ***Transmarking***

In some applications, it may be useful to convert auxiliary information embedded in a media signal from one format to another. This converting process is referred to as transmarking. Transmarking may include converting an out of band identifier like a tag in a header/footer to a watermark or vice versa. It may also involve converting a message in one watermark format to another. The process involves a decoding operating on an input media object, and an encoding of the decoded information into the media object. It may also involve a process for removing the mark originally in the input object to avoid interference with the newly inserted mark.

There are a variety of reasons to perform transmarking. One is to make the embedded information more robust to the types of processing that the media object is likely to encounter, such as converting from one watermark used in packaged media to another watermark used in compressed, and electronically distributed media, or a watermark used in radio or wireless phone broadcast transmission applications.

This type of transmarking process may be performed at various stages of a media object's distribution path. As suggest previously, an identifier in a watermark or file header/footer may be encoded at the time of packaging the content for distribution, either in an electronic distribution format or a physical packaged medium, such as an optical disk or magnetic memory device. At some point, the media signal may be converted from one format to another. This format conversion stage is an opportunity to perform transmarking that is tailored for the new format in terms of robustness and perceptibility concerns. The new format may be a broadcast format such as digital radio broadcast, or AM or FM radio broadcast. In this case, the identifier may be transmarked into a watermark or other metadata format that is robust for broadcast applications. The new format may be a compressed file format (e.g., ripping from an optical disk to an MP3 format). In this case, the identifier may be transmarked into a

file header/footer or watermark format that is robust and compatible with the compressed file format.

The transmarking process may leave an existing embedded identifier in tact and layer an additional identifier into the media object. This may include encoding a new watermark that does not interfere with an existing watermark (e.g., insert the new watermark in unmarked portions of the media object or in a non-interfering transform domain). It may also include adding additional or new identifier tags to headers or footers in the file format.

#### 10 ***Amplifying an Embedded Identifier***

Rather than converting embedded data to another format, an amplifying process may be used to renew an identifier that has become weakened or separated due to processing of the media object in which it is embedded. In this case, a decoder and encoder pair may be used to determine the current identifier and re-encode it. Of course, the encoder can also choose to embed a new or additional identifiers as well.

If the previous identifier is lost, the encoder can query an identifier database established in the registration process, passing identifying information about the media object. The database uses the identifying information to find an associated identifier and returns it to the encoder for embedding in the media object.

#### 20 ***Managing On-Line Media Library Through Links In Media Signals***

The forms in which digital media content can be distributed continue to evolve rapidly. Video and audio signals can be stored in a digital content package and distributed in physical form, such as an optical or magnetic storage medium, or in an electronic form (e.g., transferred over a network in a compressed or uncompressed form). In this document, a content package refers to a format in which a title, e.g., a film, song, musical album, multimedia collection etc., is played from a complete representation of that title.

In contrast, media content may also be delivered over a wire or wireless communication link in a streaming format. Obviating the need to have a complete copy of the title, a streaming format enables the receiver to play the title as it receives portions of it in a data "stream" from an external source. The following sections

- 25 -

describe applications for linking media signals to other content and data using metadata and/or steganography.

***Linking Packaged Digital Media to On-Line Library of Media Titles***

- In this application, a local application (e.g., a device or software process)
- 5 extracts an identifier from a media signal stored in a content package, and communicates the identifier to a database application to create and manage a library of media titles. Examples of a content package include optical media such as CDs and DVDs, magnetic media such as floppy disks and tapes, flash memory, compressed media files, etc. The user places the package into a media reader, such as a disk drive,
- 10 player, etc. Operating in conjunction with the media reader, the local application extracts information (e.g., a portion of the media signal) from the package, extracts the identifier, and sends it to a database system (e.g., a server on the Internet). In response, the database system determines the corresponding title and adds the title to an on-line library (e.g., external storage accessible via the Internet). The library may be set up as
- 15 a personal collection, or a collection for a group of users.

To identify the user(s)' library, the local application provides a user identifier. This user identifier may be authentication information entered by a user (such as a user name and password), or alternatively, may be an identifier (such as a device ID) sent automatically by the local application.

- 20 The title (i.e. content) is added to the on-line library, by transferring a copy of the selection (e.g., music track, video, etc.) from a master database (e.g., a library of MP3 files, or some other streaming or downloadable content format) to the user's on-line library collection. This arrangement avoids the need to upload content from the user's application. Also, it is a much more secure approach than techniques that simply
- 25 read title data from a CD and relay same to the on-line library. (It is a simple task for an unscrupulous user to fake the presence of a CD by determining how the client CD software specifies the title to the on-line library, and then mimic same even without possession of a bona fide CD.) The in-band encoding presented by watermarks offers innately better security, and provides opportunities for enhanced security by encryption,
- 30 etc.

- 26 -

In other arrangements, a copy of the selection, per se, is not transferred from the master database to the user's library, but rather a reference (e.g., a link or pointer) to the master library is added to the user's library. Efficiencies in storage can thereby be achieved (i.e., a copy of each selection is stored only once, from which an unlimited number of users' on-line libraries can link to it).

The identifier may be placed in the content package by steganographically encoding it in the media signal. For example, the identifier may be a reference number (e.g., of 24 – 256 bits) or the text name of the title embedded in a digital watermark. In a digital watermark implementation, a watermark embedder encodes the identifier in video, audio and/or images. The local application includes a watermark detector that reads at least a portion of the media signal from the package, detects the watermark, and reads the identifier embedded in the watermark. The detector may be implemented in a computer program (e.g., driver application, browser plug-in, etc.). A communication application, such as an Internet browser, then communicates the identifier to the database system, which may be implemented using conventional database management and Internet server software.

One advantage of this application is that it allows a user to create an on-line library of titles, and then playback those titles from the library on demand. For example, the user may organize a large collection of titles, view titles in a variety of formats, and playback individual songs or videos, in any order and at any time. The user can request playback anywhere by connecting to the on-line database and requesting a streaming delivery or file download.

For playback, a player application (e.g., device or application program on a computer) sends a request to a content delivery system via a wire or wireless connection. The content delivery system first checks to make sure that the user has the title in her on-line library. In addition, it may authenticate the user and determine usage rights before returning any content. If it determines playback to be authorized, the content delivery system sends the titles by streaming the content to the player application, on demand, in the order requested.

***Linking Streaming Media to On-Line Library of Media Titles***

A similar scheme to the one described in the previous section may be implemented for streaming media. In this case, the local application need not have a packaged version of the content to add a title to a user's library. Instead, the local application extracts an identifier from a portion of the streaming content. The identifier may be embedded in a watermark that is replicated throughout the media signal. In the event that the portion of the streaming media does not contain an identifier, the local application continues to execute a detection process on the media signal as it arrives until it has extracted the identifier.

In either of the above applications, the user can initiate a process of extracting the watermark by an explicit request, such as by clicking on the visual UI of the local application, entering a voice command, etc. Alternatively, the local application may initiate the detection process automatically whenever the user starts playback from packaged or streaming content.

The identifier may also include usage rights that dictate how the user (as identified by a user ID) may retrieve a copy from the library for playback. For example, the watermark may include a number that represents the number of times the user can access the content for playback.

***Linking Packaged or Streaming Media to Database of Auxiliary Information Related to the Media***

In addition to linking to a title database, the identifier may also link to other information or machine instructions relating to the media. For example, the database may send a set of options back to the user (e.g., in the form of a HTML page) that allow the user to select and download additional information related to the media signal in which the identifier is embedded.

***Operating Environment for Computer Implementations***

Figure 3 illustrates an example of a computer system that serves as an operating environment for software implementations of the systems described above. The software applications may be implemented in C/C++ and are portable to many different computer systems. Fig. 3 generally depicts one such system.

- 28 -

The computer system shown in Fig. 3 includes a computer 1220, including a processing unit 1221, a system memory 1222, and a system bus 1223 that interconnects various system components including the system memory to the processing unit 1221.

The system bus may comprise any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using a bus architecture such as PCI, ISA and EISA, to name a few.

The system memory includes read only memory (ROM) 1224 and random access memory (RAM) 1225. A basic input/output system 1226 (BIOS), containing the basic routines that help to transfer information between elements within the computer 1220, such as during start-up, is stored in ROM 1224.

The computer 1220 further includes a hard disk drive 1227, a magnetic disk drive 1228, e.g., to read from or write to a removable disk 1229, and an optical disk drive 1230, e.g., for reading a CD-ROM or DVD disk 1231 or to read from or write to other optical media. The hard disk drive 1227, magnetic disk drive 1228, and optical disk drive 1230 are connected to the system bus 1223 by a hard disk drive interface 1232, a magnetic disk drive interface 1233, and an optical drive interface 1234, respectively. The drives and their associated computer-readable media provide nonvolatile storage of data, data structures, computer-executable instructions (program code such as dynamic link libraries, and executable files), etc. for the computer 1220.

Although the description of computer-readable media above refers to a hard disk, a removable magnetic disk and an optical disk, it can also include other types of media that are readable by a computer, such as magnetic cassettes, flash memory cards, digital video disks, and the like.

A number of program modules may be stored in the drives and RAM 1225, including an operating system 1235, one or more application programs 1236, other program modules 1237, and program data 1238.

A user may enter commands and information into the personal computer 1220 through a keyboard 1240 and pointing device, such as a mouse 1242. Other input devices may include a microphone, joystick, game pad, satellite dish, digital camera, scanner, or the like. The microphone may be used to capture audio signals. Similarly, a digital camera or scanner 43 may be used to capture video and images. The camera and scanner are each connected to the computer via a standard interface 44. Currently,

- 29 -

there are digital cameras designed to interface with a Universal Serial Bus (USB), Peripheral Component Interconnect (PCI), and parallel port interface. Two emerging standard peripheral interfaces for cameras include USB2 and 1394 (also known as firewire and iLink).

5           These and other input devices are often connected to the processing unit 1221 through a serial port interface 1246 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port or a universal serial bus (USB).

10           A monitor 1247 or other type of display device is also connected to the system bus 1223 via an interface, such as a video adapter 1248. In addition to the monitor, personal computers typically include other peripheral output devices (not shown), such as speakers and printers.

          The computer 1220 operates in a networked environment using logical connections to one or more remote computers, such as a remote computer 1249. The  
15           remote computer 1249 may be a server, a router, a peer device or other common network node, and typically includes many or all of the elements described relative to the computer 1220, although only a memory storage device 1250 has been illustrated in Figure 3. The logical connections depicted in Figure 3 include a local area network (LAN) 1251 and a wide area network (WAN) 1252. Such networking environments are  
20           commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

          When used in a LAN networking environment, the computer 1220 is connected to the local network 1251 through a network interface or adapter 1253. When used in a WAN networking environment, the personal computer 1220 typically includes a  
25           modem 1254 or other means for establishing communications over the wide area network 1252, such as the Internet. The modem 1254, which may be internal or external, is connected to the system bus 1223 via the serial port interface 1246.

          In a networked environment, program modules depicted relative to the personal computer 1220, or portions of them, may be stored in the remote memory storage device. The processes detailed above can be implemented in a distributed fashion, and  
30           as parallel processes. It will be appreciated that the network connections shown are exemplary and that other means of establishing a communications link between the computers may be used.

- 30 -

The computer may establish a wireless connection with external devices through a variety of peripherals such as a cellular modem, radio transceiver, infrared transceiver, etc.

- While a computer system is offered as example operating environment, the applications may be implemented in a variety of devices and systems, including servers, workstations, hand-held devices (e.g., hand held audio or video players, Personal Digital Assistants such as Palm Pilot, etc.), network appliances, distributed network systems, etc.

### 10 ***Managing Local and Remote Collections of Media Objects***

- In this application, media object links, such as unique content identifiers, facilitate the creation of media object systems that enable user's to manage local and remote collections of media objects. The local collection is maintained at the end-user's machine (e.g., player, computer, multimedia terminal, etc.), and includes media objects (songs, movies, etc.) along with links to remote collections of the media objects, or alternatively, only links to the remote collections. The remote collection includes a collection of media objects in a central or distributed database on one or more server computers accessible via a communication link (e.g., computer network, telephone, satellite, cable, etc.). The remote collection may be structured so that individual users (e.g., subscribers) each maintain a personal list of media objects, and these lists of media objects refer to a central or distributed database where the actual copies of the media objects are stored.

- In one application for music, the user has a local database system that enables the user to create play lists of songs in the local database. After selecting a list of songs for a play list from within the local collection, the user synchronizes the play list with a personal collection maintained at a remote system. To synchronize, the local database on the user's computer sends a list of media object identifiers to the remote system along with the name of the play list. Later, the user can access that play list from a portable player. The user plugs in the portable player to a network connection, and accesses the music in the play list from the remote system, e.g., via streaming delivery or file download. This approach enables the user to set up play lists with a computer



- 31 -

using a keyboard and large screen, and then retrieve the play list on the computer or a portable audio player.

In addition, if the local database also stores the copies of the songs in a secure format, the user can transfer the songs to a portable player or portable storage device, and listen to them on the computer, portable player, home stereo system, or car stereo without connecting to the remote system. Later, when the computer or player is connected to the remote system, it sends identifiers of the songs that have been transferred to the device and/or played, to the remote system so that fees can be attributed to the user's account and royalties calculated. If the user discontinues use of the system (e.g., cancels a subscription), the remote system disables the local database and optionally deletes it. The songs in the local database are stored in a secure format (e.g., encrypted) that requires the user to have a key to play them or transfer them to a portable player. It is preferable not to delete the local database so that the user can re-subscribe later with minimal effort. The media objects may also be linked to advertisements so that the remote system can include advertising content (audio or graphical displays) along with the copies of the songs. This advertising can be played during playback of the related songs.

The local collection may include only links to the songs in the remote system, or include copies of the songs locally as well as links to the songs in the remote system. The local database can be implemented by making a local secure music container with links to the songs in the remote system. Alternatively, individual copies of songs in the local database can be encrypted along with links to these files in the remote system. The songs may not need to be packaged in secure containers like encrypted files if the user owns a copy of the song (as in the case where the user owns the CD).

When connected to the remote system, the local database system on the user's computer synchronizes the songs in the user's local database with the remote system. When the user adds songs to his personal collection on the remote system, possibly through online purchases, a content identification process is used to identify the song for the local database. This content identification process may identify the content in a variety of ways as described elsewhere in this document. Some examples include decoding a digital watermark carrying a content ID embedded in the song, deriving a digital audio signature (e.g., ID dynamically derived from content or fingerprint), or

- 32 -

extracting a metadata tag from the content file or secure digital container (e.g., an encrypted file that encapsulates the song content and metadata). When songs are added to the local database, possibly through physical music CD or MP3 purchases, the same happens. The watermark is implicitly secure while it is optimal if metadata tags (such as contained in ID3 tags for artist, album and song title) are digitally signed using well-known encryption techniques to verify security. In addition, the data packet sent between the remote and local databases can be secured using well-known encryption technology for secure authenticating channels.

## 10 **Concluding Remarks**

Having described and illustrated the principles of the technology with reference to specific implementations, it will be recognized that the technology can be implemented in many other, different, forms. Such modifications are within the scope of the present invention. Processes and components described in this application may be used in various combinations, and in some cases, interchangeably with other processes and components described above. Of course, the particular combinations of elements and features in the above-detailed embodiments are exemplary only.

- 33 -

We claim:

1. A method of linking a media object to metadata or actions comprising:  
in a user's player device, decoding an identifier from the media object during or  
before playback of the object;  
5 forwarding the identifier to a remote server via a communication link;  
in the remote server, mapping the identifier to an action;  
executing the action, including returning data to the player device and  
presenting at least a portion of the data to a user during or before playback.
- 10 2. The method of claim 1 wherein the identifier is in a file header or footer of  
the media object.
3. The method of claim 1 wherein the identifier is derived from the media  
object.
- 15 4. The method of claim 1 wherein the identifier is derived from metadata stored  
on packaged media that stores the media object.
5. The method of claim 4 wherein the packaged media is an optical disk.
- 20 6. The method of claim 1 wherein the identifier is a broadcaster or distributor ID  
and context information is used to identify the media object.
7. The method of claim 6 wherein the context information includes a  
25 timestamp, the broadcaster ID is used to identify a corresponding broadcaster playlist,  
and the timestamp is used to look up a media object broadcast at the time indicated by  
the timestamp.
8. The method of claim 1, further comprising:  
30 forwarding context information along with the identifier; and

- 34 -

using the context information along with the identifier to map the identifier to an action, such that a different action is executed in response to the identifier based on the context information.

5           9. The method of claim 8, wherein the context information identifies a type of distribution such that a different action is executed based on the type of distribution.

10           10. The method of claim 8, wherein the context information identifies a type of device such that a different action is executed based on the type of device.

11. The method of claim 8, wherein the context information identifies information about a user such that a different action is executed based on the information about the user.

15           12. The method of claim 1, wherein the identifier is used to look up data that is returned to the player device and controls rendering of the media object.

13. The method of claim 12, wherein the identifier and context data are used to look up data that controls rendering of the media object.

20           14. The method of embedding a media object with an identifier, including:  
transferring the media object from a packaged medium;  
obtaining an identifier based on identifying information of the media object;  
coding the media object into an electronic file format, including inserting the  
25 identifier in the electronic file format; wherein the identifier is used to link the media object to metadata or an action.

15. The method of claim 14 wherein the media object comprises an audio signal.

30           16. The method of claim 14 wherein the media object is an audio signal transferred from an optical disk to a compressed file format.

17. The method of claim 14 wherein the identifier is derived from the media object or metadata about the media object stored on the packaged medium.

- 5           218. The method of claim 14 further including:  
            presenting a user interface enabling a user to purchase or rent a link for the media object, where the link associates the media object with metadata or an action provided by the user and the link is effected by associating the identifier with the user provided metadata or action.

- 10           219. A computer readable medium on which is stored software for performing the method of claim 14.

220. A wireless telephone handset including a recorder for recording a media  
15    signal transmitted to the handset, and for recording ambient audio received via a microphone.

221. The wireless handset of claim 20 further comprising:  
            a decoder for decoding an identifier from at least a portion of a media signal  
20    recorded with the recorder; and  
            a transmitter for transmitting the identifier to a server operable to map the identifier to an action or metadata associated with the media signal.

222. The handset of claim 20 wherein the identifier is an out of band signal  
25    associated with the media signal.

            223. The handset of claim 20 wherein the identifier is an in-band signal embedded into the media signal.

- 30           224. The handset of claim 20 including a display device for displaying metadata returned from the server.

- 36 -

25. The handset of claim 20 including a display device for displaying metadata extracted from the media signal.

26. The handset of claim 25 wherein the display device is operable to display  
5 metadata items extracted for two or more media objects received in the handset,  
the handset including a user input device for enabling the user to select from  
among the displayed metadata items; wherein the handset is responsive to the user  
selection to request an action or additional metadata about the selected media object  
from a remote source.

10

27. A method of establishing an on-line collection of media titles, comprising:  
extracting an identifier steganographically encoded in a media signal;  
sending the identifier to a database; and  
requesting the database to add a title associated with the media signal to an  
15 line collection.

28. The method of claim 27 further including:  
requesting playback of a title from the on-line collection.

20

29. A method of managing a collection of media titles comprising:  
enabling a user to create a list of media objects in a local database; and  
synchronizing the list with the user's on-line media object collection in a remote  
system by sending content identifiers for the media objects in the playlist to the remote  
system.

25

30. The method of claim 29 wherein the remote system associates the list with  
the user such that the user can retrieve the media objects in the list via a communication  
link to a player device.

30

31. The method of claim 29 including:  
tracking media objects played on a device of the user by keeping a record of  
content identifiers of media objects played on the device; and

- 37 -

when the device is connected to the remote system, sending the content identifiers to the remote system to keep track of the media objects that the user has played.

- 5           32. A system which manages a collection of media titles comprising:  
a database including a list of media objects;  
an interface to a remote system having a media collection stored therein; and  
a synchronization module to synchronize the database with the remote system  
via  
10 a list of media object identifiers.

33. The method of claim 32, further comprising an interface through which the list of media object identifiers is transferred to a portable device.

- 15           34. A system to interact with an on-line collection of media titles stored in a database, said system comprising:  
a detector to extract an embedded watermark from a media signal, the watermark including identifying information for the media signal;  
a module to send the identifying information to the database; and  
20 a module to receive data from the database in response to the identifying information.

35. The system according to claim 34, wherein the identifying information includes a title of the media signal to be added to the on-line collection.

- 25           36. The system according to claim 34, wherein the identifying information includes usage rights.

37. The system according to claim 34, wherein the media signal is stored in a  
30 content package.

38. An apparatus to interact with a media signal, said apparatus comprising:

- 38 -

a storage device;  
a processing unit; and  
a local application stored in said storage device and processed by said processing unit, said local application operating to: i) extract an identifier from the media signal, and to ii) send the identifier to a database to associate the media signal with an on-line collection maintained by the database.

39. The apparatus according to claim 38, wherein the identifier comprises a watermark.

10

40. The apparatus according to claim 38, wherein the identifier comprises a reference number.

41. The apparatus according to claim 38, wherein the identifier comprises data embedded within a watermark.

15

42. The apparatus according to claim 41, wherein the data comprises a title name.

43. The apparatus according to claim 38, wherein said local application provides a user identifier to the database.

20

44. The apparatus according to claim 38, wherein said user identifier comprises a device identifier.

25

45. The apparatus according to Claim 38, wherein the media signal is stored in a content package.

46. A method of operating a system to create and manage a library of on-line media titles, said method comprising the steps of:  
receiving a media identifier from a user;

30



- 39 -

determining a title that corresponds to the media identifier;  
identifying the user; and  
adding the media title to an on-line library of media titles associated with the  
user.

5

47. The method according to claim 46, further comprising the step of:  
in response to a user's request for a media title, verifying that the media title is  
in  
the user's on-line library.

10

48. The method according to claim 47, further comprising the steps of:  
authenticating the user in response to the user's request for a media title; and  
allowing the user to access the on-line library when the user is authenticated.

15

49. The method according to claim 48, further comprising the steps of:  
determining usage rights associated with a requested media title, and controlling  
access to the media title based on the usage rights.

20

50. The method according to claim 46, wherein said adding step comprises the  
step of transferring a copy of the media title to the on-line library.

51. The method according to claim 46, wherein said adding step comprises the  
step of adding a pointer to the on-line library to point to a copy of the media title.

25

52. The method according to claim 46, wherein the user is identified by  
identifying a user device in communication with the system.

53. An apparatus to interact with a streaming media signal, said apparatus  
comprising:

30

a storage device;  
a processing unit; and

- 40 -

a local application stored in said storage device and processed by said processing unit, said local application operating to: i) extract an identifier from the streaming media signal, and to ii) send the identifier to a database to associate the streaming media signal with an on-line library collection maintained by the database.

5

54. The apparatus according to claim 53, wherein the identifier is embedded in a watermark in the streaming media signal.

55. The apparatus according to claim 54, wherein the watermark is replicated  
10 throughout the streaming media signal.

56. The apparatus according to claim 54, wherein the local application extracts the identifier upon a user request.

57. The apparatus according to claim 56, wherein the request comprises a voice  
15 command.

58. The apparatus according to claim 54, wherein the identifier comprises  
usage rights which dictate how the user may retrieve a copy of the streaming media  
20 signal from the library for playback.

59. The apparatus according to claim 54, wherein the on-line library is a personal library associated with a predetermined user.

60. The apparatus according to claim 54, wherein the on-line library is  
25 associated with a predetermined group of users.

30

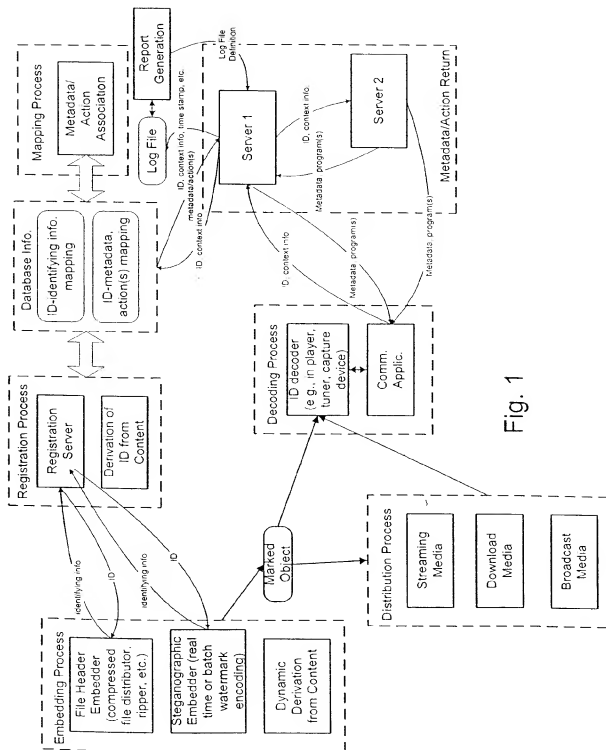


Fig. 1

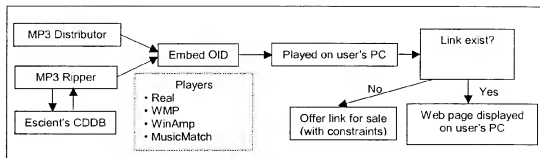
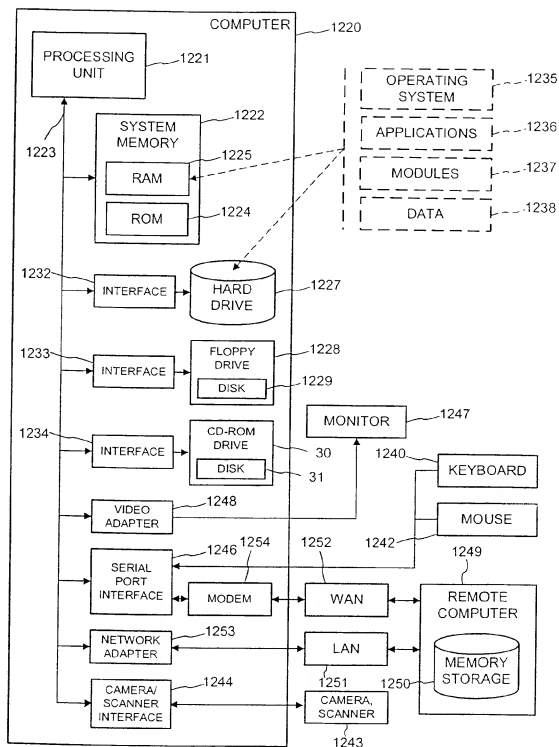


Fig. 2

Fig. 3



## INTERNATIONAL SEARCH REPORT

Internatic application No.

PCT/US01/02609

## A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 17/00; H04K 1/00  
 US CL : 382/100, 317; 455/412, 466; 709/219

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)  
 U.S. : Please See Continuation Sheet

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 97/43736 A1 (DIGIMARC CORPORATION) 20 November 1997 (20.11.1997), Figures 43, 60 and 63; page 74, line 13 through page 79, line 27; page 84, line 1 through page 86, line 28; page 121, lines 10-17; page 138, line 5 through page 141, line 31.	1-60
X	WO 99/41900 A1 (FRAUNHOFER-GESELLSCHAFT ZUR FÖRDERUNG DER ANGEWANDTEN FORSCHUNG E.V.) 19 August 1999 (19.08.1999), Figure 1, page 4, line 23 through page 8, line 29.	1-19, 27-60
X	WO 99/57623 A2 (FRAUNHOFER CENTER FOR RESEARCH IN COMPUTER GRAPHICS, INC.) 11 November 1999 (11.11.1999), Figures 9, 10; page 22, line 27 through page 27, line 28.	1-19, 27-60
X	US 5,319,453 A (COPRIVIZA et al.) 07 June 1994 (07.06.1994), Figures 1, 17; Abstract; column 3, line 63 through column 5, line 40; column 7, line 66 through column 10, line 34.	1-19, 27-60
X	US 5,875,249 A (MINTZER et al.) 23 February 1999 (23.02.1999), Figure 1; column 6, line 18 through column 7, line 5.	1-19
X	US 5,841,978 A (RHOADS) 24 November 1998 (24.11.1998), column 57, line 39 through column 60, line 34.	1-19

☒ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

<p>* Special categories of cited documents:</p> <p>"A" document defining the general state of the art which is not considered to be of particular relevance</p> <p>"E" earlier application or patent published on or after the international filing date</p> <p>"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>"O" document referring to an oral disclosure, use, exhibition or other means</p> <p>"P" document published prior to the international filing date but later than the priority date claimed</p>	<p>"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention</p> <p>"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone</p> <p>"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art</p> <p>"Z" document member of the same patent family</p>
---	---

Date of the actual completion of the international search

Date of mailing of the international search report

25 March 2001 (25.03.2001)

06 JUN 2001

Name and mailing address of the ISA/US  
 Commissioner of Patents and Trademarks  
 Box PCT  
 Washington, D.C. 20231  
 Facsimile No. (703)305-3230

Authorized officer

Andrew W. Johns

Telephone No. (703) 305-3900

Form PCT/ISA/210 (second sheet) (July 1998)

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/US01/02609

## C (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 649 074 A1 (XEROX CORPORATION) 19 April 1995 (19.04.1995), Figures 1, 2, 7, column 4, line 14 through column 5, line 15; column 8, line 13 through column 13, line 5.	1-19
X	ZHAO, J., "A WWW Service to Embed and Prove Digital Copyright Watermarks," Proc. European Conf. on Multimedia Applications, Services and Techniques, May 1996, see sections 2, 4, 5.	1-19
X	VOYATZIS, G. et al., "The Use of Watermarks in the Protection of Digital Multimedia Products," Proc. IEEE, vol. 87, no. 7, July 1999, see sections IV-VI on pages 1201-1205.	1-19
X	DEFOVERE, G. et al., "The VIVA Project: Digital Watermarking for Broadcast Monitoring," IEEE Proc. Int. Conf. on Image Processing, October 1999, see section 1: Introduction on pages 202-203.	1-19

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/US01/02609

Continuation of B. FIELDS SEARCHED Item 1: 382/100, 232, 305, 306, 317; 705/51; 707/104; 709/ 217, 218, 219;  
713/176, 179; 380/210, 252, 287; 379/88.11, 88.12, 88.13, 88.17, 908; 455/412, 466, 556; D14/138, 141, 144